CLAIMS

What is claimed is:

1. A graphics processing method, comprising the steps of:
(a.) caching texture memory fetches using a cache tag assignment which is essentially unique mapped, while
(b.) generating condensed cache tags by combining
a mip-mapping-level-of-detail parameter which can have at least $2^{J-1}+1$ different values
together with
coordinate bits which can have as many as 2^K different values
into fewer than J+K bits
without loss of information
(c.) and using said condensed tags for said caching step (a.).

2. The method of Claim 1, wherein said step (c) exploits an interrelationship between the number of possible values of said coordinate bits for some values of said mip-mapping-level-of-detail parameter.

3. A graphics processing method, comprising caching texture memory fetches using a cache tag assignment in which a unique relation defines a smaller tag address for any given memory address.

4. The graphics processing method of Claim 3, wherein said cache tag assignment is generated by combining a mip-map-level-of-detail parameter which can have at least $2^{J-1}+1$ different values together with coordinate bits which can have as many as 2^K different values into fewer than J+K bits without loss of information.

5. The graphics processing method of Claim 3, wherein said cache tag assignment is generated by combining a first parameter which can have at least $2^{J-1}+1$ different values together with coordinate bits which can have as many as 2^K different values into fewer than J+K bits without loss of information;

wherein said first parameter and said coordinate bits are three-dimensional coordinates.

6. A method of generating condensed cache tags, comprising the steps:

(a.) concatenating the texel address on the x- and y-axis with a map level identifier, where addresses on the x-axis can require **m** bits, addresses on the y-axis can require **n** bits, and said map-level identifier can require **p** bits;

(b.) if two caches are being used for odd/even maps, deleting the least significant bit of said map level identifier;

(c.) if texels are being stored in the cache in $2^i x 2^j$ patches, deleting the **i** least significant bits of the address on the x-axis and the **j** least significant bits of the address on the y-axis;

(d.) coding said map level identifier so that

the largest map level uses 1 bit to designate the map level and ((**m**-**i**)+(**n**-**j**)) bits to specify said addresses on said x- and y-axis,

the second largest map level uses 3 bits to designate the map level and ((**m**-**i**)+(**n**-**j**)-2) bits to specify said addresses on said x-axis and y-axis, and

successively smaller map levels use greater than 3 bits to designate the map level and less than ((**m**-**i**)+(**n**-**j**)-2) bits to specify said addresses on said x-axis and y-axis.

7. A cache system for a texture map, comprising:

a texture memory containing at least one map, wherein the addresses for said map can require **m** bits for the x-axis, **n** bits for the y-axis, and **p** bits for the map-level identifier;

a direct-mapped texture cache for said texture memory wherein a lookup tag requires **m**+**n**-1 or fewer bits.